IN THE UNITED STATES PATENT AND TRADEMARK OFFICE BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

In re Application of: Batalden et al.	§	Group Art Unit: 2173
Serial No.: 10/020,026	§	
Confirmation No.: 8987	§	Examiner: Ting Zhou
Filed: December 14, 2001	§	
For: Method For Controlling A Browser Session	§	

MAIL STOP APPEAL BRIEF - PATENTS
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

CERTIFICATE OF MAILING OR TRANSMISSION

I hereby certify that this correspondence is being deposited with the United States Postal Service with sufficient postage as first class mail in an envelope addressed to: Mail Stop Appeal Brief - Patents, Commissioner for Patents, P. O. Box 1450, Alexandria, VA 22313-1450, or facsimile transmitted to the U.S. Patent and Trademark Office to fax number 571-273-8300 to the attention of Examiner Ting Zhou, on the date shown below:

March 12, 2007	/Jon K. Stewart/
Date	Jon K. Stewart

RESPONSE TO NOTICE OF NON-COMPLIANT APPEAL BRIEF

Dear Sir:

In response to the Notice of Non-Compliant Appeal Brief, Appellants submit this Substitute Appeal Brief to the Board of Patent Appeals and Interferences in response to the Notification of Non-Compliant Appeal Brief dated February 8, 2007 in the Appeal of the above-identified application.

TABLE OF CONTENTS

Real Party in Interest

The present application has been assigned to International Business Machines Corporation, Armonk, New York.

Related Appeals and Interferences

Applicant asserts that no other appeals or interferences are known to the Applicant, the Applicant's legal representative or assignee which will directly affect or be directly affected by or have a bearing on the Board's decision in the pending appeal.

Status of Claims

Claims 1-4, 7, 9-15, 17, 20-26, 28, and 30-36 are pending in the application. Claims 1-33 were originally presented in the application. Claims 5, 6, 16, 18, 19, 27, and 29 have been canceled without prejudice. Claims 1-4, 7, 9-15, 17, 20-26, 28, and 30-36 stand finally rejected as discussed below. The final rejections of claims 1-4, 7, 9-15, 17, 20-26, 28, and 30-36 are appealed. The pending claims are shown in the attached Claims Appendix.

Status of Amendments

All claim amendments have been entered by the Examiner, including amendments to the claims proposed after the final rejection.

Summary of Claimed Subject Matter

Claimed embodiments include methods (see e.g., claims 1-4, 7, 9-13, 34-36), computer programs stored on computer readable storage media (see e.g., claims 14, 15, 17, 20-24) and computer systems (see e.g., claims 25, 26, 28, 30-33) for controlling the operation and appearance of browsers. And more particularly, for using a controlling browser window to control a controlled browser window by overriding a predetermined response to user input received by the controlled browser window. Generally, an action specified by an event handler may be used to override the response to user input; and instead, cause a response different from the predetermined response.

One claimed embodiment includes a method (see e.g., claim 1) for controlling a viewable browser window. *See Application*, *Abstract*, 3:25-29, 5:15-23. Claim 1 recites the step of opening a controlling browser window configured to control aspects of a controlled browser window. *See Application*, 8:11-31, 10:32-11:2, 13:23-30, 14:4-5, Figure 6, method 600, steps 620, 622. Also, Figure 4, 400 and 400 and Figure 5, 502, and 510 provide an example of an interface that includes an example of a controlling browser window and a controlled browser window. As claimed, the controlling browser window establishes at least one event handler prior to opening the controlled browser window. *See Application*, 13:17-21. During a browsing session, the event handlers serve to monitor user input and produce a response specified by the event handler. *See Application*, 8:3-9, 8:22-31, 15:4-6. For an example illustration of an event handler, *see also Application*, 12:10-34.

Claim 1 also recites the step of opening the controlled browser window, wherein the controlled browser window includes a display area for rendering viewable content received from network locations. *See Application*, 5:17-18, 14:4-13. As claimed, the controlling browser window controls at least one functional aspect of the controlled browser window during a browsing session engaged in by a user. *See Application*, 8:11-31, 15:4-6, 15:22-27 (an example of controlling the browsing functionality of the controlled browser window), 16:7-17:5 (describing the control of the event handlers over a variety of functional aspects of the controlled browser window).

Claim 1 also recites the step of receiving user input to which the controlled browser window is configured to produce a predetermined response. *See Application*, 15:4-6, 16:7-17:5. The final step of claim 1 recites overriding the predetermined response by executing an action specified by the at least one event handler to cause a response different from the predetermined response. *See Application*, 8:18-24, 15:22-28 (describing the overriding of the response to the user input of "file:" into the address filed of the controlled browser window) and 16:18-24 (describing the overriding a user request to open a network location and instead rendering contend from a specified URL in the controlled browser window). *See also, Application* 16:26-34, 17:1-5.

Another claimed embodiment (see e.g., claim 14) includes a computer readable storage medium containing a controlling browser program which, when executed, performs operations for controlling a controlled browser window comprising a display area for rendering viewable content received from network locations. See *Application*, *Abstract*, 4:1-5, 5:15-23, 25-34. As claimed, the operations include opening a controlling browser window configured to control aspects of the controlled browser window by establishing at least one event handler prior to opening the controlled browser window, and wherein the controlling browser window controls at least one functional aspect of the controlled browser window, during a browsing session engaged in by a user. *See Application*, 8:22-31, 13:17-21, 15:4-6. For an example illustration of an event handler, *see Application* 12:10-34. *See also Application*, 8:11-31, 15:4-6, 15:22-27 (an example of controlling the browsing functionality of the controlled browser window) and 16:7-17:5 (describing the control of the event handlers over a variety of functional aspects of the controlled browser window).

As claimed, the operation also includes receiving user input to which the controlled browser window is configured to produce a predetermined response. *See Application*, 5:17-18, 14:4-13. The operation also includes overriding the predetermined response by executing an action specified by the at least one event handler to cause a response different from the predetermined response. *See Application*, 8:18-24, 15:22-28 (describing the overriding of the response to the user input of "file:" into the address field of the controlled browser window), 16:18-24 (describing the overriding a user request to open a network location and instead rendering contend from a specified URL in the controlled browser window). *See also Application,* 16:26-34, 17:1-5.

Another claimed embodiment (see e.g., claim 25) includes a computer system having a memory containing at least a browser programming (see Application, 7:31-34. And a processor (see Application, 8:11-15) which when executing the browser programming is configured to open a controlling browser window configured to control aspects of a controlled browser window by establishing at least one event handler prior to opening the controlled browser window, the aspects comprising at least one operational aspect of a graphical user interface. See Figures 4, 402, 5, 5210 showing operational aspect of a graphical user interface. *See also Application*, 8:22-31, 13:17-21, 15:4-6. For an example illustration of an event handler, *see Application* 12:10-34. *See Application*, 8:11-31, 15:4-6, 15:22-27 (an example of controlling the browsing functionality of the controlled browser window) and 16:7-17:5 (describing the control of the event handlers over a variety of functional aspects of the controlled browser window).

As claimed, the browser program is also configured to open the controlled browser window, wherein the controlled browser window includes a display area for rendering viewable content received from network locations, and wherein the controlling browser window controls the at least one operational aspect of the controlled browser window during a browsing session engaged in by a user. *See Application*, 5:17-18, 14:4-13. *See Application*, 8:11-31, 15:4-6, 15:22-27 (an example of controlling the operational aspects of the controlled browser window) and 16:7-17:5 (describing the control of the event handlers over a variety of functional aspects of the controlled browser window).

As claimed, the browser program is also configured to receive user input to which the controlled browser window is configured to produce a predetermined response and override the predetermined response by executing an action specified by the at least one event handler to cause a response different from the predetermined response. *See Application*, 15:4-6, 16:7-17:5. *See Application*, 8:18-24, 15:22-28 (describing the overriding of the response to the user input of “file:” into the address field of the controlled browser window), 16:18-24 (describing the overriding a user request to open a network location and instead rendering contend from a specified URL in the controlled browser window). *See also Application*, 16:26-34, 17:1-5.

Another claimed embodiment (see, e.g., claim 34) includes a method of controlling a browsing session engaged in by a user. The method includes opening a browser program that opens a first browser window, configured to open and display an HTML document, wherein the HTML document includes an executable component, which when processed by the browser program causes the browser program to establish at least one event handler, and further configured to open a second browser window. *See Application*, 8:11-31, 10:1-11, 10:32-11:2, 13:23-30, 14:4-5, Figure 6, method 600, steps 620, 622. Also, Figure 4, 400 and 400 and Figure 5, 502, and 510 provide an example of an interface that includes an example of a first browser window and a second browser window.

As claimed, this method recites the step of opening the second browser window, wherein the executable component of the first browser window is further configured to control at least functional aspect of the second browser window, during the browsing session engaged in by a user interacting with the second browser window. *See Application*, 8:22-31, 13:17-21, 15:4-6. For an example illustration of an event handler, *see Application*, 12:10-34. *See also*, *Application*, 8:11-31, 15:4-6, 15:22-27 (an example of controlling the browsing functionality of the controlled browser window), 16:7-17:5 (describing the control of the event handlers over a variety of functional aspects of the controlled browser window). This method also recites the step of receiving user input to which the second browser window is configured to produce a predetermined response and overriding the predetermined response by executing an action specified by the at least one event handler to cause a response different from the predetermined response. *See Application*, 15:4-6, 16:7-17:5. *See Application*, 8:18-24, 15:22-28 (describing the overriding of the response to the user input of “file:” into the address field of the controlled browser window) 16:18-24 (describing the overriding a user request to open a network location and instead rendering contend from a specified URL in the controlled browser window). *See also, Application* 16:26-34, 17:1-5.

Grounds of Rejection to be Reviewed on Appeal

1. Claims 14-15, 17, 20-24 stand rejected under 35 U.S.C. § 112, first paragraph.

2. Claims 1-2, 4, 7, 12-15, 20, 25-26, 28, 32-34, and 36 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over *Yen et al.,* (U.S. Pub. No. 20020054141) in view of *Hodgkinson* (U.S. Pub. No. 20020016802).

3. Claims 3, 9-11, 17, 21-24, 30-31 and 35[1] stand rejected under 35 U.S.C. § 103(a) as being unpatentable over *Yen*, *Hodgkinson*, in further view of Netscape® Communicator 4.75, copyright 2000.

.

[1] [1] A Notice of Non Compliant Appeal brief dated, February 8, 2007 provides: "Under the third heading (issue), claim 19 was omitted from the list of claims that are rejected in view of *Yen* and *Hodgkinson* and *Netscape*. Please correct." However, claim 19 was cancelled during prior prosecution in this matter. Nevertheless, the Examiner's final rejection continued to reject this cancelled claim. Applicants have removed references to claim 19 from this Appeal Brief.

ARGUMENTS

Claims 14-15, 17, 20-24 and 35 U.S.C. § 112, first paragraph.

Claims 14-15, 17 and 20-24 are rejected under 35 U.S.C. § 112, first paragraph, as failing to comply with the written description requirement. More specifically, the Examiner suggested that an amendment to claim 14 made in response to a § 101 rejection amounted to the addition of new matter. In response to this rejection, Applicants amended claim 14 and discussed the 35 U.S.C. § 112, first paragraph rejection on pages 8 and 9 of Applicants' response to the Examiner's Final Office Action. The Examiner entered the after-final amendment, but did not comment about this rejection in the Advisory Action in this matter. Thus, the status of this rejection is somewhat unclear. Applicants contend, however, that the amendments entered after-final address the original § 101 rejection without running afoul of the proscription against the addition of new matter under § 112.

Obviousness of Claims 1-2, 4, 7, 12-15, 20, 25-26, 28, 32-34, and 36 over *Yen* in view of *Hodgkinson*

The Applicable Law

The Examiner bears the initial burden of establishing a *prima facie* case of obviousness. *See* MPEP § 2142. To establish a *prima facie* case of obviousness three basic criteria must be met. First, there must be some suggestion or motivation, either in the references themselves or in the knowledge generally available to one ordinary skill in the art to modify the reference or to combine the reference teachings. Second, there must be a reasonable expectation of success. Finally, the prior art reference (or references when combined) must teach or suggest all the claim limitations. *See* MPEP § 2143. Applicants submit the present rejection fails to establish at least the first and third criteria.

The References

Yen is directed to a method for displaying multiple window displays "adjacently and simultaneously." See *Yen*, Abstract, ¶ 3, 15. *Yen* discloses that an application may display multiple windows where the display of one particular window may be dependent upon another. See *Yen*, ¶ 29. In the preferred embodiment disclosed by *Yen*, the display includes nine windows arranged in a tic-tac-toe like grid. See *Yen*, ¶ 26. When a user selects to close an application that is currently displaying some of the display windows from the grid, *Yen* discloses that the status of each window may be saved using a "display parameter." This parameter specifies whether the window is currently being displayed. See *Yen*, ¶ 38, 43. When the application is subsequently executed, "it displays all window displays 30 on the graphical user interface display 31 having display parameter 40 setting as visible." See *Yen*, ¶ 39. Importantly, as disclosed in *Yen*, the application program displaying the multiple window displays responds to user input to show (or remove) any of the windows presented on a screen in a manner consistent with the user's expectations. In other words, the application program always performs the expected action in response to the user's selection of windows to display. For example, Figure 3A shows an "extend function button 33C" used to display additional windows on the tic-tac-toe like grid. When a user clicks this button, an additional window may be displayed or an existing window may be "folded" underneath other existing window displays. See *Yen*, ¶ 32-33.

In paragraphs ¶ 46-52*Yen* describes another example configuration for a graphical user interface, one that displays stock quotes. *See Yen*, Figures 6A-6B. As shown in these figures, a button on a "first main window 46" is used to open a "first sub window display 52." Once the "first sub window display 52" is opened, the functional aspects of the "first sub window display 52" operate normally, without any control exerted by the "first main window 46." For example, the window in figure 6B includes "program function buttons" labeled 54a, 54b. *Yen*, ¶ 50. The "first main window" does not control the functional aspects of these buttons. In other words, the "first main window 46" fails to exert any control, influence, or dominion over the functional aspects or features provided by "first sub window display 52;" rather, the for the duration the "first sub window display 52" is on a display screen, this functional aspects of this window (e.g., the functional aspects of buttons 54a and 54b) continue to provide whatever functions or features these elements are configured to do.

Hodgkinson is directed to "an improved method for the generation of pages for an internet website on a display screen and particularly, although not necessarily exclusively, to the generation of pages using apparatus with relatively limited processing capability." *Hodgkinson*, ¶ 1. For example, *Hodgkinson* discloses that "Personal Computer (PC) based web browser facilities tend to search for and fetch the data for a selected internet web page" and then typically "display the display generated from the fetched data as soon as they can." *Hodgkinson*, ¶ 2. In some cases, this can result in "jerky and unreliable navigation for the user." *Hodgkinson*, ¶ 3. See *Hodgkinson*, Abstract, ¶ 1, 3. To avoid this problem, *Hodgkinson* discloses that when a user selects a web-page for display, the page is only reformatted only after specified time periods, or only after a specified amount of data has been downloaded and processed. This is done to help prevent the display of a user-selected web-page in a manner to reduce "flicker" or a "jerky" appearance, particularly on devices that lack substantial processing capability. See *Hodgkinson*, ¶ 3. *Hodgkinson* fails to disclose overriding the predetermined response that occurs when a user selects a webpage to download and view, instead *Hodgkinson* alters the conventional way in which a web page is formatted and reformatted.

Despite the focus in both *Yen* and *Hodgkinson* on techniques for the arrangement and presentation of data in windows on a display screen, the Examiner asserts that *Yen* teaches a method for controlling a viewable browser window that includes opening a controlling browser window configured to control aspects of a controlled browser window, wherein the controlling browser window establishes at least one event handler prior to opening the controlled browser window and opening the controlled browser window, … wherein the controlling browser window controls at least one functional aspect of the controlled browser window during a browsing session engaged in by a user, as recited by claim 1. Claims 14, 25, and 34 each recite a similar limitation.

On this point, the Examiner relies on paragraphs 46-52 from *Yen* described above. However, once the "first sub window display 52" is opened, the "first main window 46" (which the Examiner asserts is analogous to the "controlling browser window" of the present claims) fails to control a "functional aspect of the controlled browser window." Instead, functional aspects of the "first sub window display 52" (which the Examiner asserts is analogous to the "controlled browser window") operate free from influence or control of the "first main window 46." Accordingly, Applicants submit that *Yen* fails to disclose the limitations of a controlling browser window configured to control aspects of a controlled browser window, as recited by the present claims.

Further, Applicants submit that the combination of *Yen* and *Hodgkinson* fails to disclose a "controlling browser window" and "controlled browser window," configured to "receive[] user input to which the controlled browser window is configured to produce a predetermined response" and to "override[] the predetermined response by executing an action specified by the at least one event handler to cause a response different from the predetermined response," as recited by claims 1, 14, 25, and 34. The Examiner concedes that *Yen* fails to teach these limitations but suggests that the Combination of Yen in view of *Hodgkinson* renders these limitations obvious. Specifically, the Examiner asserts:

> *Hodgkinson* further teaches receiving user input to which the controlled browser window is configured to produce a predetermined response and overriding the predetermined response by executing an action to cause a response different from the predetermined response (upon receiving user selection requesting a change in the layout of the displayed webpage, instead of executing the change, the system overrides the display change and prevents the browser from reformatting the pages) (Hodgkinson: paragraph 0015 on page 2).

Final Office Action, p.5. In fact, however, the material cited by the Examiner discloses that a user may select a webpage for download that is displayed on a display device. Specifically, the cited passages disclose that the formatting of the webpage may occur at predetermined times or after a predetermined amount of data has been downloaded.

Absent from these passages, however, is any change whatsoever in response to the user's request to download a webpage; the only modification, so far as taught in

Hodgkinson, is when to reformat the display*. See, e.g., Hodgkinson*, ¶ 8, 15, 32, 35 and 36. Applicants submit that deferring the reformatting of a webpage as it is being download fails to disclose the recited limitation of receiving user input to which the second browser window is configured to produce a predetermined response and overriding the predetermined response, as recited by the present claims.

Far from it; claim 1 recites a step of receiving user input to which the controlled browser window is configured to produce a predetermined response and overriding the predetermined response by executing an action specified by the at least one event handler to cause a response different from the predetermined response. Claims 14, 25, and 34 recite a similar limitation. In the case (*Hodgkinson*) relied upon by the Examiner to support the rejection, the user input is a request to retrieve and render a selected web page. In response, *Hodgkinson* discloses a technique used for responding to the user input to retrieve and render the selected web-page, albeit in a manner more effectively than conventional methods. Quite plainly, however, using either conventional techniques or the technique disclosed in *Hodgkinson*, the request and response are identical; namely, to retrieve and render the selected web-page.

Furthermore, the Examiner suggests that the combination of *Yen* and *Hodgkinson* would "provide a management system that keeps users from conducting harmful or unauthorized actions on data, maintaining data integrity and security." See Final Office Action, p. 6. However, *Yen* "provides a computer system for displaying multiple windows adjacently and simultaneously." See *Yen*, Abstract. And *Hodgkinson* "relates to the generation of user selected pages and the delay in the reformatting of the same following a user selection … of a new page which is to be displayed. See *Hodgkinson*, Abstract. Both cases are directly related to how information is displayed, as such, these references alone, or combined, do not disclose any features at all related to a management system "that keeps users from conducting harmful or unauthorized actions on data, maintaining data integrity and security." Accordingly, Applicants contend that the proffered motivation to combine these references to "data integrity and security" is fundamentally flawed as material relied upon by the Examiner from both *Yen* and *Hodgkinson* has nothing to do with data integrity or data security. Moreover, it's unclear how techniques for the arrangement and display of information would enhance data integrity or data security in any event.

Regarding claims 2, 4, 7, 12, 13, 15, 20, 26, 28, 32, 33, and 36, each of these claims depends from one of claims 1, 14, 25, or 34. Applicants believe the above remarks demonstrate that *Yen* in view of *Hodgkinson* fails to disclose each and every limitation of the independent claims. Therefore, Applicants believe these dependent claims are allowable over *Yen* in view of *Hodgkinson*.

For all of the foregoing reasons, withdrawal of this rejection and allowance of claims 1-2, 4, 7, 12-15, 20, 25-26, 28, 32-34, and 36 is respectfully requested.

Obviousness of Claims 3, 9-11, 17, 21-24, 30-31 and 35 over *Yen* in view of *Hodgkinson* and *Netscape*[2]

Each of these rejected claims depends from one of independent claims 1, 14, 25 and 34. As set forth above, *Yen*, in view of *Hodgkinson*, fails to disclose a method for controlling a viewable browser window that includes receiving user input to which a second browser window is configured to produce a predetermined response and overriding the predetermined response by executing an action specified by the at least one event handler to cause a response different from the predetermined response.

Furthermore, claims 3, 9-11, 17, 21-24, 30-31 and 35 recite limitations directed browser chrome elements that may be included with the controlling browser window. For example, claim 3 recites specifies that the "controlling browser window is further configured to control a graphical aspect of the controlled browser including the control of at least one browser chrome element displayed by a graphical user interface displayed by the opened controlled browser window." Claims 9-11, 17, 21-24, 30-31 and 35 include similar limitations.

The Examiner concedes that *Yen* fails to teach "aspects of the controlled browser window to be controlled by the controlling browser window [that] comprise at least one browser chrome element displayed by a graphical user interface displayed by the opened controlled browser window." *See Final Office Action*, p. 10. The Examiner

[2] See Footnote 1, page 11.

asserts that *Yen*, in view of Hodgkinson and *Netscape* renders the limitations recited by these claims obvious.

Applicants contend that *Netscape* does not teach or suggest a controlling at least one browser chrome element displayed by a graphical user interface displayed by the opened controlled browser window. In rejecting claims these claims, the Examiner cites screenshots taken of the *Netscape* program being used to display a web-page. As shown, the screenshots includes a user selecting a link on one of the screenshots to open a second browser window. In particular, the link to the second browser window is activated using a context sensitive menu choice: "Open Link in New Window." However, once the second window is opened, (screenshot 3 of the *Netscape* program) the original browser window (screenshot 2 of the *Netscape* program) fails to exert any control or influence over the second browser window, and fails to control any chrome element displayed by the opened controlled browser window. More generally, the original browser window fails to control any visual aspect of the controlled browser window. The user remains free to engage in any action or invoke any feature provided by the *Netscape* program, without any restriction or control effectuated by the first browser window. Further, in processing the "open link in new window" menu selection, the first browser window (screenshot 2) fails to exert any control over the visual appearance or functionality of the second browser window (screenshot 3).

The Examiner argues that the controlled browser window (presumably, the window of figure 2) is "controlling at least one browser chrome element by deactivating the display of a portion of the chrome, i.e., some of the navigation buttons such as the "back" and "forward" buttons." *See Office Action*, p 7. Respectfully, this argument mischaracterizes the *Netscape* screenshots. In this example, the "back" and "forward" buttons in screenshots 3 are not deactivated, and nor has the functional operations of these buttons modified in any way, by a controlling browser window, or otherwise; rather, the buttons appear "grayed out" because there are no "forward" or "back" browsed pages to access. Quite plainly, the visual aspects of the *Netscape* browser window are not controlled by a controlling browser window. Thus, the combination of *Yen* and *Netscape* fail to teach or suggest at least one browser chrome element displayed by a graphical user interface displayed by the opened controlled browser window. Accordingly, Applicants submit that claims 3, 9-11, 17, 21-24, 30-31 and 35 are allowable, and respectfully request, therefore, that the rejection be withdrawn and the claims be allowed.

For all of the foregoing reasons, withdrawal of this rejection and allowance of claims 3, 9-11, 17, 21-24, 30-31 and 35 is respectfully requested.

CONCLUSION

The Examiner errs in finding:

- Claims 14-15, 17, 20-24 fail to satisfy the written description requirement under 35 U.S.C. § 112, first paragraph.
- Claims 1-2, 4, 7, 12-15, 20, 25-26, 28, 32-34, and 36 are unpatentable over *Yen* in view of *Hodgkinson* under 35 U.S.C. § 103(a).
- Claims 3, 9-11, 17, 21-24, 30-31 and 35 are unpatentable over *Yen*, *Hodgkinson*, in further view of *Netscape* under 35 U.S.C. § 103(a).

Withdrawal of these rejection and allowance of all claims is respectfully requested.

Respectfully submitted, and
S-signed pursuant to 37 CFR 1.4,

/Gero G. McClellan, Reg. No. 44,227/
Gero G. McClellan
Registration No. 44,227
Patterson & Sheridan, L.L.P.
3040 Post Oak Blvd. Suite 1500
Houston, TX 77056
Telephone: (713) 623-4844
Facsimile: (713) 623-4846
Attorney for Appellant(s)

CLAIMS APPENDIX

1. (Previously Presented) A method for controlling a viewable browser window, comprising:

opening a controlling browser window configured to control aspects of a controlled browser window, wherein the controlling browser window establishes at least one event handler prior to opening the controlled browser window; and

opening the controlled browser window, wherein the controlled browser window includes a display area for rendering viewable content received from network locations, and wherein the controlling browser window controls at least one functional aspect of the controlled browser window during a browsing session engaged in by a user;

receiving user input to which the controlled browser window is configured to produce a predetermined response; and

overriding the predetermined response by executing an action specified by the at least one event handler to cause a response different from the predetermined response.

2. (Original) The method of claim 1, wherein the viewable content is Web content.

3. (Previously Presented) The method of claim 1, wherein the controlling browser window is further configured to control a graphical aspect of the controlled browser including the control of at least one browser chrome element displayed by a graphical user interface displayed by the opened controlled browser window.

4. (Previously Presented) The method of claim 1, wherein opening the controlling browser window further comprises, locking at least one of a keyboard key and a mouse key.

5-6. (Canceled)

7. (Previously Presented) The method of claim 1, further comprising:

re-establishing the at least one event handler for each change in a network address being accessed by the opened controlled browser window.

8. (Canceled).

9. (Previously Presented) The method of claim 1, wherein opening the controlling browser window comprises preventing at least a portion of chrome of the opened controlled browser window from being displayed on an output device.

10. (Previously Presented) The method of claim 9, wherein the chrome of the opened controlled browser window comprises at least one of a tool bar, a menu bar, a title bar, an address field, and a border.

11. (Original) The method of claim 9, wherein the controlling browser window comprises at least one of a tool bar, a menu bar, a title bar, an address field, and a border.

12. (Original) The method of claim 1, wherein opening the controlled browser window comprises executing a controlled browser program and wherein opening the controlling browser window comprises executing a controlling browser program.

13. (Previously Presented) The method of claim 12, further comprising, in response to receiving user input configured to produce a first action by the opened controlled browser program, executing the controlling browser program to override the first action and produce a second action.

14. (Previously Presented) A computer readable storage medium containing a controlling browser program which, when executed, performs operations for controlling a controlled browser window comprising a display area for rendering viewable content received from network locations, the operations comprising:

opening a controlling browser window configured to control aspects of the controlled browser window by establishing at least one event handler prior to opening the controlled browser window, and wherein the controlling browser window controls at least functional aspect of the controlled browser window, during a browsing session engaged in by a user;

receiving user input to which the controlled browser window is configured to produce a predetermined response; and overriding the predetermined response by executing an action specified by the at least one event handler to cause a response different from the predetermined response.

15. (Original) The computer readable medium of claim 14, wherein the viewable content is Web content.

16. (Canceled)

17. (Previously Presented) The computer readable medium of claim 14, wherein the controlling browser window is further configured to control a graphical aspect of the controlled browser window including the control of at least one browser chrome element displayed by a graphical user interface displayed by the opened controlled browser window.

18-19. (Canceled)

20. (Original) The computer readable medium of claim 14, wherein opening the controlling browser window comprises locking at least one of a keyboard key and a mouse key.

21. (Previously Presented) The computer readable medium of claim 14, wherein the controlling browser window is configured to restrict the browsing activity engaged in by the user by limiting access to at least one network address that is accessible by the opened controlled browser window.

22. (Previously Presented) The computer readable medium of claim 14, wherein opening the controlling browser window comprises preventing at least a portion of chrome of the opened controlled browser window from being displayed on an output device.

23. (Previously Presented) The computer readable medium of claim 22, wherein the chrome of the opened controlled browser window comprises at least one of a tool bar, a menu bar, a title bar, an address field, and a border.

24. (Original) The computer readable medium of claim 22, wherein the controlling browser window comprises at least one of a tool bar, a menu bar, a title bar, an address field, and a border.

25. (Previously Presented) A computer, comprising:

a memory containing at least a browser programming;

a processor which when executing the browser programming, is configured to:

open a controlling browser window configured to control aspects of a controlled browser window by establishing at least one event handler prior to opening the controlled browser window, the aspects comprising at least one operational aspects of a graphical user interface; and

opening the controlled browser window, wherein the controlled browser window includes a display area for rendering viewable content received from network locations, and wherein the controlling browser window controls the at least one operational aspect of the controlled browser window during a browsing session engaged in by a user;

receiving user input to which the controlled browser window is configured to produce a predetermined response; and

overriding the predetermined response by executing an action specified by the at least one event handler to cause a response different from the predetermined response.

26. (Previously Presented) The computer of claim 25, wherein the browser programming comprises a Web browser.

27. (Canceled)

28. (Previously Presented) The computer of claim 25, further comprising a keyboard and a mouse and wherein, by executing the browser programming to open the controlling browser window, the processor is configured to lock at least one of a keyboard key and a mouse key.

29. (Canceled)

30. (Previously Presented) The computer of claim 25, further comprising a display device and wherein, by executing the browser programming, the processor is

configured to prevent at least a portion of chrome of the controlled browser window from being displayed on the display device.

31. (Previously Presented) The computer of claim 30, wherein the chrome of the controlled browser window comprises at least one of a tool bar, a menu bar, a title bar, an address field, and a border.

32. (Previously Presented) The computer of claim 25, further comprising a network connection configured to support communications with the network locations via a network.

33. (Original) The computer of claim 32, wherein the network is the Internet.

34. (Previously Presented) A method of controlling a browsing session engaged in by a user, comprising:

opening a browser program that opens a first browser window, configured to open and display an HTML document, wherein the HTML document includes an executable component, which when processed by the browser program causes the browser program to establish at least one event handler, and further configured to open a second browser window;

opening the second browser window, wherein the executable component of the first browser window is further configured to control at least functional aspect of the second browser window, during the browsing session engaged in by a user interacting with the second browser window;

receiving user input to which the second browser window is configured to produce a predetermined response; and

overriding the predetermined response by executing an action specified by the at least one event handler to cause a response different from the predetermined response.

35. (Previously Presented) The method of claim 34, wherein the aspects of the second browser window to be controlled by the first browser window further comprise controlling a graphical aspect of at least one browser chrome element displayed by the second browser window.

36. (Previously Presented) The method of claim 34, wherein the executable component processed by the browser program renders the first browser window as a hidden window and the second browser window as a viewable window, and wherein the browsing activity engaged in by the user is restricted by the executable component.

EVIDENCE APPENDIX

NONE

RELATED PROCEEDINGS APPENDIX

NONE.